TP4 – Classification bayésienne

Dans ce TP, nous souhaitons résoudre un problème de classification binaire (classe C0 et C1) en utilisant la classification bayésienne. Chaque classe va être modélisée en utilisant la méthode « Kernel Density Estimation (KDE) »

I. Chargement et visualisation des données

Charger les données (TP4.npy) et visualiser les en utilisant le code donné au TP3. Quelle est la dimension des données ? Le nombre d'exemples par classe ?Peut-on modéliser les classes par une gaussienne ?

II. Estimation des densités de probabilité

Ecrire une fonction ddp_parzen($\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Cov}$) qui estime la densité de probabilité en un point $\boldsymbol{x}$ à partir des points de la base d'apprentissage $\boldsymbol{X} = \{\boldsymbol{x_i}\}$, en utilisant un noyau 2D gaussien. On utilisera une matrice de covariance diagonale avec un écart type σ identique pour les deux dimensions. On pourra utiliser la fonction norm2() développée au TP3. On rappelle que :

ddp_parzen($\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{Cov}$) = $\frac{1}{N}\sum_{i=1}^{N} norm2(\boldsymbol{x}, \boldsymbol{x_i}, \boldsymbol{Cov})$

Questions
Rappeler le principe de fonctionnement de l'estimation de densité de probabilité par noyau. Que représente le noyau ?

Estimer la densité de probabilité (ddp) pour les pixels de classe C0 $p(\boldsymbol{x}/\text{C0})$ et C1 $p(\boldsymbol{x}/\text{C1})$. Afin de visualiser ces ddp, estimer les en tout point $\boldsymbol{x} = \begin{bmatrix} x_a \\ x_b \end{bmatrix}$ d'une grille carrée $(1 \leq x_a, x_b \leq 20)$ et stocker les dans deux matrices $p0$ et $p1$ que l'on pourra visualiser sous forme d'images (attention à retourner l'image avant de la visualiser, np.flipud). Visualiser la ddp de chacune des deux classes et faire varier σ.

Questions
Comment varient les ddp en fonction de σ ? Etait-ce prévisible ? *A priori* et sans faire le test, quelle valeur de σ pensez-vous adéquate pour faire la classification ?

III. Classification bayésienne

Estimer les densités *a priori* de chaque classe P0 et P1.

Réaliser la classification des points $\boldsymbol{x}$ de la base de test en utilisant la règle de bayes :

$$C = arg\max_i\{p(Ci/\boldsymbol{x})\} = arg\max_i\{\text{ddp\_parzen}(\boldsymbol{x}, \boldsymbol{X}_i, \boldsymbol{Cov}) * \text{Pi}\}$$

Où $\boldsymbol{X}_i$ est l'ensemble des exemples d'apprentissage de la classe Ci.

Afficher le taux de reconnaissance. Faire varier σ et optimiser le taux de reconnaissance.

Questions
Pour quelle valeur de σ obtient-on la meilleure classification ? Etait-ce prévisible ? Expliquer le résultat.